CLAIMS

What is claimed is:

1. A method for executing program instructions comprising the steps of:

assigning each individual ones of a plurality of program steps a
5 unique number;

issuing a program step to an execution queue;

selecting a specific one of a plurality of numbered locations in the
execution queue based upon the unique number of the issued program step,
each numbered location having an instruction valid bit and the execution
10 queue having a certain total number of numbered locations;

determining the value of the instruction valid bit associated with the
selected numbered location;

using the certain total number and the unique number of the issued
program step, calculating value of a status bit for the selected numbered
15 location; and

based upon the determined value of the instruction value bit and the
calculated value of the status bit, determining availability of the selected
numbered location such that the issued program step is stored in the selected
numbered location if the selected numbered location is available, and a full
20 flag for the execution queue is issued if the selected numbered location is not
available.

2. The method of claim 1, wherein the unique number is a monotonically
ascending series of integers assigned to the program steps in the same order
as the program is executed.

25 3. The method of claim 1, wherein the execution queue is one of a load queue
and a store queue.

4.   The method of claim 1, wherein the program step is issued to the execution queue in an order determined by an availability of a selected one of a plurality of computation resources.

5.   The method of claim 1, wherein each program step issued to the execution queue is assigned a queue location number based upon ascending value of the unique number assigned to the program step.

6.   The method of claim 5, wherein the assigned queue location number is one of a plurality of numbered locations in each one of a plurality of execution queues and is selected by calculating and recording a modulus of the queue location number.

7.   The method of claim 6, wherein the divisor of the modulus calculation is equal to a number of locations in the queue.

8.   The method of claim 7, wherein the status bit of the selected numbered location is switched when said location becomes invalid.

9.   The method of claim 1, wherein a load queue is part of the execution queue.

10.   An apparatus for executing program instructions comprising:
          means for assigning each individual ones of a plurality of program steps a unique number;
          means for issuing a program step to an execution queue;
          means for selecting a specific one of a plurality of numbered locations in the execution queue based upon the unique number of the issued program step, each numbered location having an instruction valid bit and the execution queue having a total number of numbered locations;
          means for determining the value of the instruction valid bit associated with the selected numbered location;

means for using the total number and the unique number of the issued program step to calculate value of a status bit for the selected numbered location; and

based upon the determined value of the instruction valid bit and the calculated value of the status bit, means for determining availability of the selected numbered location in the execution queue, such that the issued program step is stored in the selected numbered location if the selected numbered location is available, and a full flag for the execution queue is issued if the selected numbered location is not available.

11. The apparatus of claim 10, further comprising a plurality of computer systems, all with access to a plurality of computation resources.

12. The apparatus of claim 10 wherein the unique number is a monotonically ascending series of integers assigned to the program steps in the same order as the program is executed.

13. The apparatus of claim 10, wherein the execution queue is one of a load queue and a store queue.

14. The apparatus of claim 10, wherein the execution queue is selected out of a plurality of execution queues based on the type of the issued program step, and the program step is issued to the execution queue in an order determined by an availability of a selected one of a plurality of computation resources.

15. The apparatus of claim 14, wherein each program step issued to the execution queue is assigned a queue location number based upon ascending value of the unique number assigned to the program step.

16. The apparatus of claim 15, wherein the assigned queue location number is one of a plurality of numbered locations in each one of the plurality of

execution queues and is selected by calculating and recording a modulus of the queue location number.

17.   The apparatus of claim 16, wherein the divisor of the modulus calculation is equal to a number of locations in the queue.

5   18.   The apparatus of claim 17, wherein the status bit of the selected numbered location is switched when said location becomes invalid.

19.   The apparatus of claim 10, wherein a load queue is part of the execution queue.